

MCMULLEN'S ROOT-FINDING ALGORITHM FOR CUBIC POLYNOMIALS

JANE M. HAWKINS

(Communicated by Michael Handel)

ABSTRACT. We show that a generally convergent root-finding algorithm for cubic polynomials defined by C. McMullen is of order 3, and we give generally convergent algorithms of order 5 and higher for cubic polynomials. We study the Julia sets for these algorithms and give a universal rational map and Julia set to explain the dynamics.

1. INTRODUCTION

In 1987 Curtis McMullen gave an answer to a question posed by Steve Smale a few years earlier on the existence of algorithms for finding zeros of polynomials ([6] and [3]). Does there exist a rational root-finding algorithm which converges to a root for almost every initial guess and for almost every polynomial? We use Lebesgue measure in the complex plane and on the space of coefficients of polynomials. An algorithm with this property is called a *generally convergent* algorithm.

By a root-finding algorithm we mean a map which associates a rational map of the Riemann sphere to each polynomial of degree d with the following property: the roots of the polynomial should be attracting fixed points of the rational map. An initial guess yields a sequence converging to a root of the polynomial (namely the forward orbit of the initial guess under the rational map resulting from applying the algorithm to the polynomial) if and only if it lies in the attracting basin of the root. Most commonly implemented root-finding algorithms are not generally convergent (see e.g. [3]). It is well known that Newton's algorithm is a generally convergent algorithm of order 2 for quadratic polynomials. Several generally convergent algorithms for quadratic polynomials with distinct roots which are of higher order are discussed in [8] and [2].

McMullen proved several important results about root-finding algorithms in [3]. First, he proved that there can be no generally convergent algorithm for polynomials of degree 4 and higher, by establishing a rigidity result on parametrized families of rational maps. In addition he gave an explicit formula for a generally convergent algorithm for cubic polynomials. If complex conjugation is used in the algorithm, then generally convergent root-finding algorithms have been given by Shub and Smale [5] for polynomials of all degrees and of n variables.

Received by the editors January 11, 2001.

2000 *Mathematics Subject Classification*. Primary 37F10, 37D20; Secondary 49M99.

Key words and phrases. Root-finding algorithms, complex dynamics.

©2002 American Mathematical Society

In this paper we discuss McMullen's algorithm, the Julia sets it produces when applied to cubic polynomials, and we give higher order generally convergent algorithms for cubic polynomials based on the same idea. While McMullen has already pointed out that his algorithm coincides with Newton's algorithm applied to a non-polynomial rational map related to the original polynomial, we show that it also coincides with a classical algorithm called Halley's algorithm (or König's algorithm), applied to one specific polynomial.

By viewing it as coming from Halley's algorithm however, as well as by direct proof, we show that it is a cubically convergent algorithm. Newton's algorithm has convergence of order 2 near simple roots of polynomials, but one consequence of our result is that Newton's method gives better than quadratic convergence when applied to a very special class of nonpolynomial maps. We extend the ideas used to produce McMullen's algorithm to define a different generally convergent algorithm which we show to be of order 5. We also show that there cannot be any generally convergent methods which are of order precisely 4, but arbitrarily high orders of convergence can be obtained by generally convergent algorithms for cubic polynomials.

The author thanks Kyle Kneisl for helpful discussions on the subject of root-finding algorithms, and gratefully acknowledges the referee for suggesting improvements in the paper. The graphics were produced with Mathematica programs written by the author.

2. PRELIMINARIES ABOUT ROOT-FINDING ALGORITHMS

We call two cubic polynomials p and q *similar* if there is an affine map of the form $\psi(z) = az + b$, $a, b \in \mathbb{C} - \{0\}$, such that $p(z) = q \circ \psi(z)$ for all z . It is classic that every cubic polynomial is similar to one with its quadratic term suppressed, and that every cubic polynomial that is not similar to z^3 can be given an affine change of coordinates and multiplication by a constant to be of the form $p_c(z) = z^3 + (c-1)z - c$, $c \in \mathbb{C}$. The roots of this polynomial are 1 , $\frac{1}{2}(-1 - \sqrt{1-4c})$, and $\frac{1}{2}(-1 + \sqrt{1-4c})$. With this particular parametrization of cubic polynomials, 1 is always a root and the three roots add up to zero. If $c \neq \frac{1}{4}$, then the roots are distinct. For the rest of this paper (since we allow ourselves to throw out a set of measure 0 from the space of polynomials), we assume that p_c has distinct roots.

We identify $Poly_3$ with \mathbb{C} endowed with Lebesgue measure m without any loss of generality. By Rat_d we will denote the space of rational maps of the Riemann sphere \mathbb{C}_∞ of degree d . A *rational root-finding algorithm* is a map $S : Poly_3 \rightarrow Rat_d$ such that the roots of p_c are attracting fixed points of $S(p_c)$. Throughout, we will define $S_c = S(p_c)$ for all rational root-finding algorithms applied to cubic polynomials. Given a rational algorithm S , it follows that for $z \in \mathbb{C}$ in a set of positive measure, $S_c^n(z)$ converges to a root of p_c as $n \rightarrow \infty$; this is just due to the classical local form of S_c near an attracting fixed point (cf. [1]). All algorithms discussed here will be rational root-finding algorithms and referred to simply as algorithms. Given a rational map S_c obtained from an algorithm S applied to $p_c \in Poly_3$, we say that S_c is *convergent* if for m a.e. $z \in \mathbb{C}$, the sequence $S_c^n(z)$ converges to a root of p_c . Note that for a convergent map S_c , the roots of p_c must be the only attracting periodic orbits. An algorithm S is *generally convergent* if S_c is convergent for m -a.e. polynomial p_c .

An algorithm S is an *order k algorithm* ($k \geq 2$) if for every polynomial p_c , the simple roots of p_c are superattracting fixed points of S_c of order k . By this we mean that the set of fixed points of S_c includes each root r of p_c , and each root of p_c is a critical point of S_c of valency k , i.e., $S_c^{(i)}(r) = 0, i = 1, \dots, k - 1$, but $S_c^{(k)}(r) \neq 0$. We denote by $S_c^{(i)}$ the i^{th} derivative of the map S_c , while S_c^i means the i^{th} iterate.

Examples. (1) The well-known Newton algorithm for finding roots of any polynomial p of degree $d \geq 2$ is given by

$$N_p(z) = z - \frac{p(z)}{p'(z)}.$$

It is known that Newton's algorithm is at least quadratically convergent for all polynomials with simple roots. Suppose the roots of p are $\{r_1, \dots, r_d\}$. Then clearly N_p has degree d and $N_p(r_i) = r_i$. At each simple root r_i , $p'(r_i) \neq 0$. The critical points of p are the only poles of N_p , so $N_p'(r_i) = \frac{p(r_i)p''(r_i)}{(p'(r_i))^2} = 0$. One can check directly that $N_p''(r_i) \neq 0$.

- (2) It has been known since Cayley's work that Newton's algorithm is generally convergent for quadratic polynomials, but not for cubic polynomials. Indeed one can produce polynomials for which the algorithm has an attracting period 2 orbit.
- (3) Another algorithm that is cubically convergent but not generally convergent for cubic polynomials is due to Halley (cf. [2]). Also called König's method [8], it is defined by

$$H_c(z) = z - \frac{2p_c(z)p'_c(z)}{2(p'_c(z))^2 - p_c(z)p''_c(z)}.$$

One can also find "bad" cubic polynomials for Halley's algorithm by finding values of c for which a period 2 attracting orbit exists. Any small perturbation of the coefficient results in a polynomial with the same properties.

Generally convergent algorithms for quadratic polynomials are characterized in [3], and many examples are given in [8] and [2]. McMullen gives the unique degree 4 generally convergent algorithm for cubic polynomials. In the next section we study its connection to Halley's algorithm. The following theorem motivates our study of generally convergent algorithms.

Theorem 2.1 ([3]). *Every generally convergent algorithm for cubic polynomials is obtained by specifying a rational map \mathcal{M}_1 such that:*

- (1) \mathcal{M}_1 is convergent for $p_1(z) = z^3 - 1$;
- (2) its centralizer $C(\mathcal{M}_1)$ contains the Möbius transformations that permute the cube roots of unity.

The algorithm is then given (for p_c with distinct roots) by

$$\mathcal{M}_c = \phi_c \circ \mathcal{M}_1 \circ \phi_c^{-1},$$

where ϕ_c is a Möbius transformation carrying the cube roots of unity to $1, \frac{1}{2}(-1 - \sqrt{1 - 4c}),$ and $\frac{1}{2}(-1 + \sqrt{1 - 4c})$ in any order.

We say that a rational map R generates a generally convergent algorithm (for cubic polynomials) if R satisfies the hypotheses of Theorem 2.1. A simple inspection reveals that the map from Halley's algorithm $H_1(z) = \frac{z(2+z^3)}{1+2z^3}$ satisfies the necessary and sufficient conditions to generate a generally convergent algorithm. Namely it

commutes with the symmetric group on the cube roots of unity; clearly $H_1(1/z) = 1/H_1(z)$ and $H_1(\zeta z) = \zeta H_1(z)$ where ζ is any cube root of unity. If H_1 is convergent, we can apply Theorem 2.1 to obtain a generally convergent algorithm. This is shown below and is a classical result.

3. McMULLEN'S ALGORITHMS FOR CUBIC POLYNOMIALS

McMullen gives a formula for a generally convergent algorithm for a cubic polynomial, which we will denote by \mathcal{M} , in terms of the coefficients of the polynomial p_c . For the polynomial $p_1(z) = z^3 - 1$, the formula yields the map

$$\mathcal{M}_1(z) = \frac{z(2 + z^3)}{1 + 2z^3}.$$

Our first observation is that this is identical to the Halley algorithm map for the same polynomial. However instead of extrapolating to other polynomials as in the equation for H_c above (since we just discussed that Halley's algorithm is not generally convergent for cubics), we simply conjugate \mathcal{M}_1 to every other \mathcal{M}_c by composing with a Mobius map that takes the 3 roots of p_1 to the 3 roots of p_c . Since the degree is 4, by uniqueness this is McMullen's algorithm.

Next we show that \mathcal{M} is an order 3 algorithm in contrast to the related order 2 Newton algorithm.

Proposition 3.1. *McMullen's algorithm \mathcal{M} is a root-finding algorithm of order 3 generated by \mathcal{M}_1 .*

Proof. Each map \mathcal{M}_c is conformally conjugate to $\mathcal{M}_1(z) = z(2 + z^3)/(1 + 2z^3)$, so it is enough to prove the result for this case. The roots of the polynomial $p_1 = z^3 - 1$ are $1, \frac{-1}{2} \pm \frac{\sqrt{3}}{2}i$. A calculation verifies that the fixed points of $\mathcal{M}_1(z)$ are the three roots, 0, and ∞ . We compute that $\mathcal{M}'_1(0) = 2 = \mathcal{M}'_1(\infty)$, while the first two derivatives of \mathcal{M}_1 vanish at the roots. In addition we calculate that $\mathcal{M}'''_1(1) = 4$ and $\mathcal{M}'''_1(\frac{-1}{2} \pm \frac{\sqrt{3}}{2}i) = -2 \pm 2\sqrt{3}i$. The result follows. \square

A straightforward calculation shows that the fixed points of \mathcal{M}_c are the roots of p_c plus the two roots ω_1 and ω_2 of the quadratic polynomial $q(z) = 3(c - 1)z^2 - 9cz - (c - 1)^2$, and the derivatives at the fixed points are 0 for the roots of p_c and 2 at $\omega_i, i = 1, 2$. If $c = 1$, then 0 and ∞ are fixed points with multiplier 2, with the roots of p_1 as the only other fixed points. Furthermore the only critical points of \mathcal{M}_c are the roots of p_c , so the Fatou set of \mathcal{M}_c consists exclusively of the attracting basins of the roots of p_c . This proves, without using Theorem 2.1, that the algorithm \mathcal{M} is generally convergent.

In Figure 1 we show the Julia set for \mathcal{M}_1 with the 3 root basins in white, black, and gray for the polynomial $p(z) = z^3 - 1$. In this case we have the same Julia set for both Halley's and McMullen's algorithms. For a typical cubic polynomial we obtain different root basin pictures for different methods, though the Julia sets are mutually conformally equivalent when the McMullen algorithm is used. In Figures 3, 4 and 5 we see the root basin pictures for the cubic polynomial $p(z) = z^3 + (2i - 1)z - 2i$ using the McMullen, Halley, and Newton algorithms respectively.

3.1. Higher order generally convergent algorithms. The rigidity result in Theorem 2.1 can be used to obtain generally convergent algorithms for cubic polynomials which are of higher order than \mathcal{M}_c . We describe the rational maps which generate a generally convergent root-finding algorithm for cubic polynomials.

Theorem 3.1. *If R generates a generally convergent root-finding algorithm for cubic polynomials, then there exist constants $a_o, a_1, \dots, a_k, a_o, a_k \neq 0$, such that R is of the following form:*

$$R(z) = \frac{z(a_o + a_1z^3 + \dots + a_kz^{3k})}{a_k + a_{k-1}z^3 + \dots + a_o z^{3k}}.$$

Proof. Assume that $R = \frac{p(z)}{q(z)}$ generates a generally convergent root-finding algorithm, and that p and q are coprime. Then by Theorem 2.1 we have that $R(\zeta z) = \zeta R(z)$ for cube roots of unity ζ which implies that the exponents on the powers of z in $q(z)$ must be multiples of 3, and $p(z) = z(a_o + a_1z^3 + \dots + a_kz^{3k})$, with $a_k \neq 0$. Furthermore, since $R(\zeta 0) = R(0) = \zeta R(0)$, the points 0 and ∞ must remain fixed or change places under R . Write $q(z) = b_o + b_1z^3 + \dots + b_jz^{3j}$ (with $b_o \neq 0$ or p and q are not coprime). Assume $b_j \neq 0$; if $a_o = 0$, then 0 is fixed and a critical point, which contradicts the assumption. Since neither 0 nor ∞ can be a fixed point of multiplicity greater than 1, we must have that $j = k$.

We have by Theorem 2.1 that

$$R(z) = \frac{z(a_o + a_1z^3 + \dots + a_kz^{3k})}{b_o + b_1z^3 + \dots + b_kz^{3k}} = \frac{z(b_k + b_{k-1}z^3 + \dots + b_o z^{3k})}{a_k + a_{k-1}z^3 + \dots + a_o z^{3k}} = 1/R(1/z).$$

We conclude that $a_k = cb_o, a_1 = cb_{k-1}, \dots, a_o = cb_k$ for some nonzero c , since the zeros of each numerator and each denominator must be the same. Using the fact that $z = 1$ is a fixed point of R we have that $c = 1$. □

Corollary 3.1. *There is no generally convergent algorithm for cubic polynomials of degree 5 or 6. The algorithm $\mathcal{Q} : \mathbb{C} \rightarrow \text{Rat}_7$ generated by*

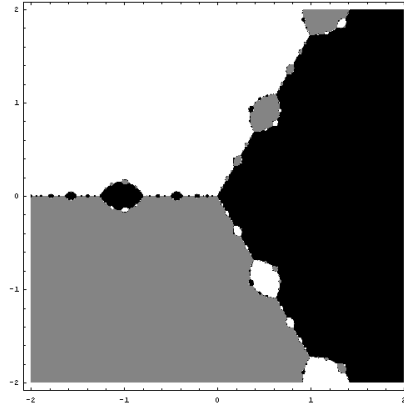
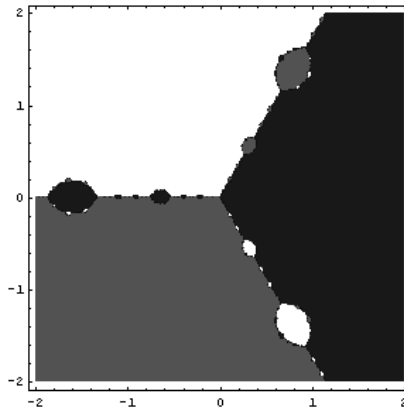
$$\mathcal{Q}_1(z) = \frac{z(14 + 35z^3 + 5z^6)}{5 + 35z^3 + 14z^6}$$

is the unique degree 7 generally convergent algorithm of order greater than 3, and it is of order 5.

Proof. We apply Theorem 3.1 to see that degree 7 is the lowest degree greater than 4 possible for a generally convergent algorithm. We consider the map $R(z) = \frac{z(a_o + a_1z^3 + a_2z^6)}{a_2 + a_1z^3 + a_o z^6}$. We assume that $a_2 = 1$ since the coefficients are defined only up to multiplication by a nonzero constant. We then solve for a_o and a_1 by setting $R'(1) = \frac{7+a_1-5a_o}{1+a_1+a_o} = 0$. This imposes the relation $a_1 = 5a_o - 7$.

Let ζ_1 and ζ_2 denote the other roots of p_1 . Since the expression for $R'(z)$ only involves powers of z^3 , $R'(\zeta_1) = R'(\zeta_2) = 0$ as well. With this relation imposed on the coefficients, we compute that $R''(1) = R''(\zeta_1) = R''(\zeta_2) = 0$, $R^{(3)}(1) = \frac{14-5a_o}{-1+a_o}$, and $R^{(4)}(1) = -6(\frac{14-5a_o}{-1+a_o})$. Since $a_o \neq 1$ (or $R(z)$ would be the identity map), the condition $R^{(3)}(1) = 0$ implies that $a_o = \frac{14}{5}$, which also implies that $R^{(3)}(\zeta_1) = R^{(3)}(\zeta_2) = R^{(4)}(1) = R^{(4)}(\zeta_1) = R^{(4)}(\zeta_2) = 0$. This follows since $R^{(4)}(z)$ only involves powers of z^3 , and $R^{(3)}(\zeta_j z) = \zeta_j R^{(3)}(z)$, $j = 1, 2$, for all z . One can check that $R^{(5)}(1) = 140/3$, so the order of the algorithm generated by $\mathcal{Q}_1 = R$ is 5. □

Using the method of Corollary 3.1 one can always solve for integer a_i 's to obtain arbitrarily high degree algorithms of arbitrarily high order. For example, using $k = 3$ in the family given in Theorem 3.1 yields the unique order 7 algorithm of degree 10 which is generated by $\mathcal{S}_1(z) = z(7 + 42z^3 + 30z^6 + 2z^9)/(2 + 30z^3 + 42z^6 + 7z^9)$. Figure 2 shows the similarity between the order 5 generally convergent algorithm

FIGURE 1. McMullen's algorithm for $p(z) = z^3 - 1$ FIGURE 2. Order 5 generally convergent algorithm for $p(z) = z^3 - 1$

and the order 3 McMullen algorithm. The order 7 algorithm \mathcal{S} gives a similar root basin picture.

We give a standard rational map as a model for the order 3 McMullen algorithm \mathcal{M} via a Möbius transformation which takes the three distinct roots of each polynomial p_c to $0, \infty$, and 1 . Similarly, each order 5 map \mathcal{Q}_c is conjugate a standard form rational map (as are all the higher order ones as well). We obtain a basic rational model for each generally convergent algorithm with bounded Julia set, from which to study the dynamics. We refer to the Julia set appearing in Figure 6 as the *universal Julia set* for the algorithm \mathcal{M} [2], since all Julia sets resulting from the algorithm \mathcal{M} are conformally equivalent to that one. The universal Julia set of \mathcal{Q} is similar as Proposition 3.3 shows.

Proposition 3.2. *For each cubic polynomial p_c with distinct roots, the map \mathcal{M}_c is conjugate to $T_3(z) = (3 + 6z^2 - z^4)/8z$, and each map \mathcal{Q}_c is conjugate to $T_5 = z(35 + 35z^2 - 7z^4 + z^6)/(8 + 56z^2)$.*

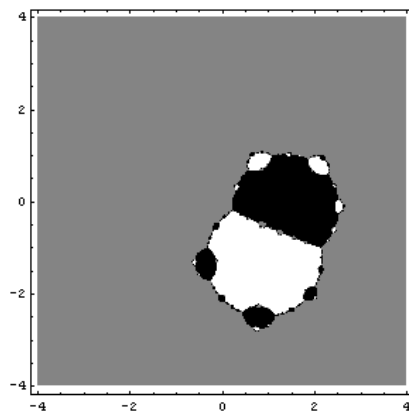


FIGURE 3. McMullen's algorithm for $z^3 + (2i - 1)z - 2i$

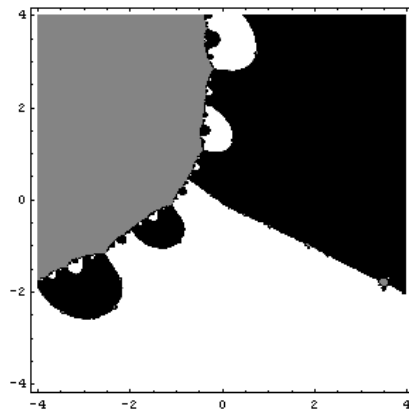


FIGURE 4. Halley's algorithm for $z^3 + (2i - 1)z - 2i$

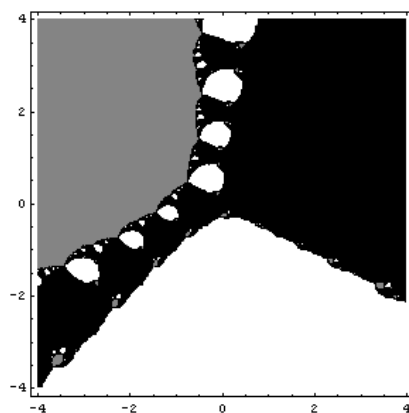


FIGURE 5. Newton's algorithm for $z^3 + (2i - 1)z - 2i$

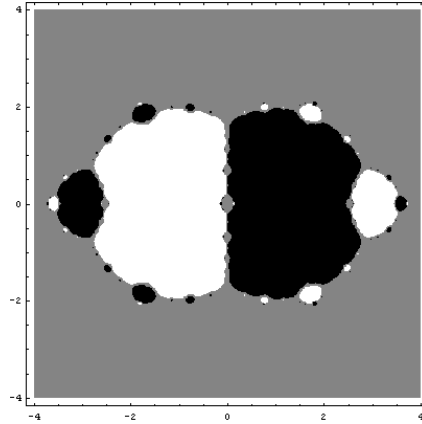


FIGURE 6. Julia set and attracting basins for $T_3(z) = \frac{3+6z^2-z^4}{8z}$

In the case of the polynomial $p_1(z) = z^3 - 1$, a conjugating map is given by

$$\phi(z) = \frac{i\sqrt{3} - (2 + e^{2\pi i/3})z}{e^{2\pi i/3}z - 1},$$

which carries the circle through the three roots of $p_1(z)$ onto the circle through $-1, 1$, and ∞ .

We list some properties of T_k and $J(T_k)$, $k = 3, 5$, and hence of McMullen's root-finding algorithm for any cubic polynomial with distinct roots. These results require little or no proof, as they are culled from the existing literature and brought together here to give some insight into the universal Julia sets that occur for generally convergent algorithms. Of course a similar theorem holds for the maps \mathcal{M}_1 and \mathcal{Q}_1 and for the higher order root finding algorithms with obvious modifications. We use these models because they have both bounded Julia sets and symmetry about the origin. In Remark 3.1 we discuss why the Julia set of T_3 (shown in Figure 6) looks like that of the algorithm \mathcal{M} applied to a typical cubic polynomial.

Proposition 3.3. *The rational maps $T_k(z)$, $k = 3, 5$, satisfy the following properties:*

- (1) *The attracting fixed points for T_k are $\infty, +1$, and -1 . Other fixed points for T_3 are $\pm\sqrt{3}i$; these are repelling with derivative 2. Other fixed points for T_5 are $\pm\sqrt{3}i$, with derivative $\frac{14}{5}$, and ± 3 and 0 , each with derivative $\frac{35}{8}$.*
- (2) *A superconvergent rational map is one with critical points only occurring at periodic points; the map T_k is superconvergent.*
- (3) *A hyperbolic rational map is one with the property that the closure of the forward orbits of all critical points is disjoint from the Julia set; T_k is hyperbolic.*
- (4) *$J(T_k)$ has Lebesgue measure 0. Hence Lebesgue almost every point converges to an attracting fixed point under T_k . For the associated root-finding algorithm, this implies that the set of initial data which lead to a root under the algorithm \mathcal{M} or \mathcal{Q} has full Lebesgue measure.*
- (5) *The immediate basin of each superattracting fixed point is simply connected, and $J(T_k)$ is connected and locally connected.*

- (6) Each component of $F(T_k)$ containing an attracting fixed point is conformally conjugate to the disk

$$\{\omega : |\omega| < |a|^{\frac{1}{1-k}}\},$$

with $a = T_k^{(k)}(1)/k!$. The local conjugating map around each fixed point extends throughout the entire component, giving cubic (or quintic) convergence to the fixed point at the center under iteration of T_k .

- (7) Consider any one of the three forward invariant Fatou components of T_k : F_∞ containing ∞ , F_1 containing 1, and F_{-1} containing -1 . Each of these is bounded by a curve.
- (8) Given any $\epsilon > 0$, there are only finitely many Fatou components of T_k with diameter greater than ϵ (using the spherical metric).

Proof. (2) follows since the critical points of T_k are all fixed points. (3) follows from (2). (4) is true since all hyperbolic maps have Julia set of measure 0 [7]. To prove (5), since T_k is superconvergent, the immediate basin of each superattracting fixed point is simply connected using [7] (Theorem 4, Chapter 3). In addition, since all critical orbits are finite, it follows that $J(T_k)$ is connected and locally connected [4]. To obtain (6) and (7), we use the superconvergence of and hyperbolicity of T_k , and (5) to apply results from [7] (Chapter 3.3, Theorem 4 and Chapter 5.5, Theorem 1). (8) is just Lemma 19.4 of [4]. □

Remark 3.1. For a cubic polynomial p_c , the Julia set of \mathcal{M}_c looks like Figure 1 if and only if $p_c = z^3 - 1$. Otherwise, the image of ∞ under the map \mathcal{M}_c is $-3c/(c-1)$. It is therefore typical that the immediate root basins are bounded for two of the three roots of p_c , and the root basin picture looks qualitatively like Figure 3 up to rescaling.

This has interesting consequences for numerical studies of McMullen's algorithms. If the roots of p_c do not all lie on a circle centered at the origin, then ∞ almost surely lies in the immediate attracting basin of one of the roots of p_c and large initial guesses will quickly converge to the root in that basin. Numerical evidence supports this statement. In Table 1 we compare the order 5 and 3 McMullen, and Newton algorithms for the polynomial $p_{2i}(z) = z^3 + (2i - 1)z - 2i$. The roots of p_{2i} are approximately 1, $-1.5643 + 0.93956i$, and $0.5643 - 0.93956i$.

TABLE 1. A numerical comparison of the algorithms \mathcal{Q}_{2i} , \mathcal{M}_{2i} , and Newton's algorithm N_{2i} .

n	\mathcal{Q}_{2i}^n	\mathcal{M}_{2i}^n	N_{2i}^n
0	1000. + 100.i	1000. + 100.i	1000. + 100.i
1	-1.73 + 0.964i	-2.4053 + 1.207i	666.67 + 66.666i
2	-1.5643 + 0.93956i	-1.5751 + 0.93746i	444.44
3	...	-1.5643 + 0.93956i	296.297 + 29.628i
.
.
19	-1.5643 + 0.93956i	-1.5643 + 0.93956i	.99553 - .00103i
20	-1.5643 + 0.93956i	-1.5643 + 0.93956i	1.00002 + .00001i

REFERENCES

- [1] A. Beardon, **Iterations of Rational Functions**, Springer-Verlag GTM 132, (1991). MR **92j**:30026
- [2] K. Kneisl, Julia sets for the supernewton method, Cauchy's method, and Halley's method, *Chaos* 11 (2001), 359-370.
- [3] C. McMullen, Families of rational maps and iterative root-finding algorithms, *Ann. of Math. Ser. 2*, 125, No.3, (1987), 467-493. MR **88i**:58082
- [4] J. Milnor, **Dynamics in One Complex Variable**, Vieweg (1999). CMP 2000:03
- [5] M. Shub and S. Smale, On the existence of generally convergent algorithms, *J. of Complexity* 2, (1986), 2-11. MR **89d**:65054
- [6] S. Smale, On the efficiency of algorithms of analysis, *Bull AMS* 13, (1985), 87-121. MR **86m**:65061
- [7] N. Steinmetz, **Rational Iteration: Complex Dynamical Systems**, de Gruyter Studies in Math 16, (1993). MR **94h**:30035
- [8] E. Vrscay and W. Gilbert, Extraneous fixed points, basin boundaries and chaotic dynamics for Schröder and König rational iteration functions, *Numer. Math.* 52, 1-16 (1988). MR **89b**:30026

DEPARTMENT OF MATHEMATICS, CB #3250, UNIVERSITY OF NORTH CAROLINA AT CHAPEL HILL, CHAPEL HILL, NORTH CAROLINA 27599

E-mail address: `jmh@math.unc.edu`